

National Aeronautics and Space Administration



# Systems Integration Challenges for a National Space Launch System

marshall



[www.nasa.gov](http://www.nasa.gov)

**Todd May**  
Space Launch System Program Manager  
*July 26, 2011*

# Expanding Humanity's Frontiers of Discovery

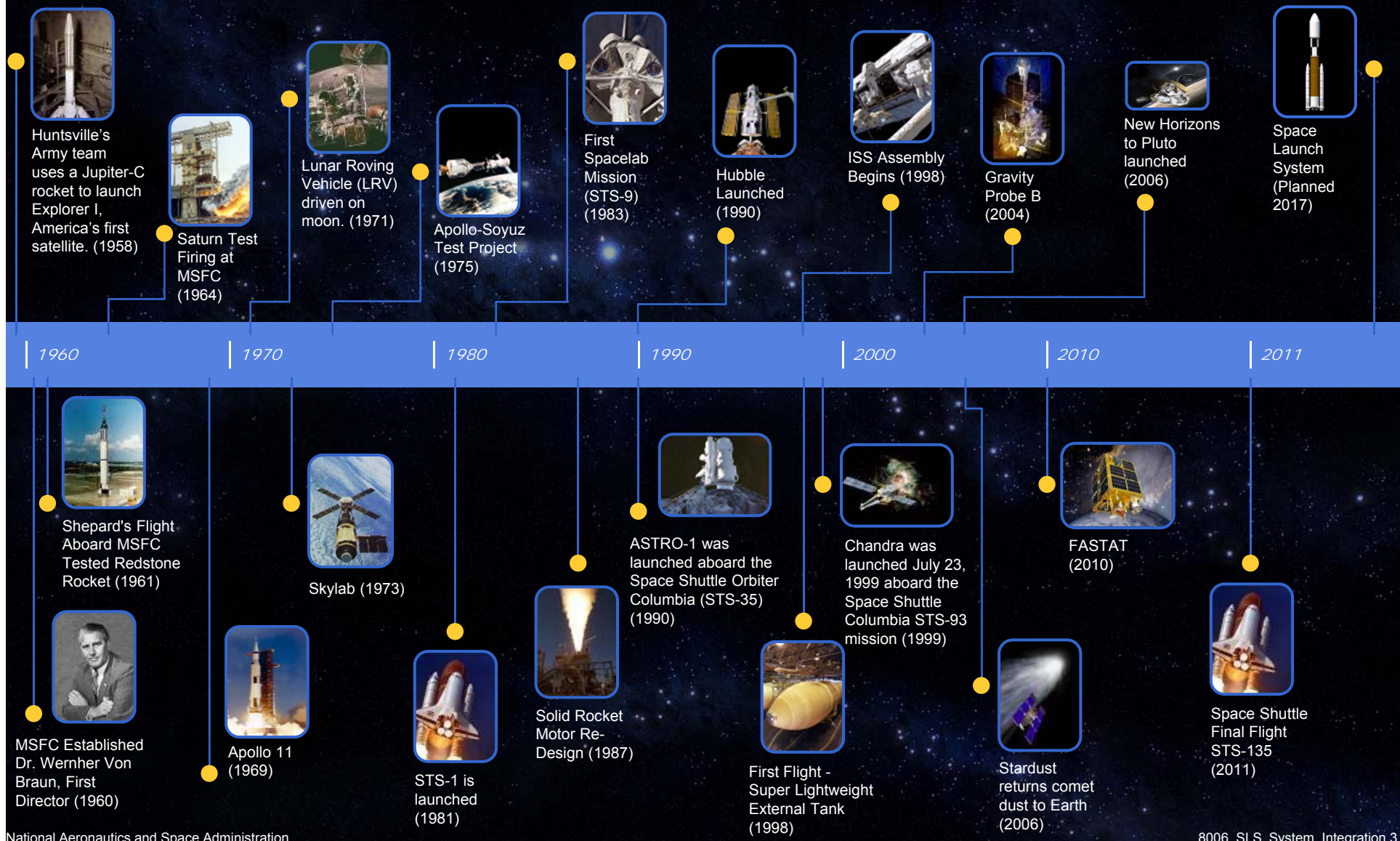




# A Brief History of the Marshall Space Flight Center (MSFC)



## Extensive Experience with Systems Engineering and Integration



# Systems Integration Development Driven by Complex Hardware and Extreme Environments



## ◆ Aerospace Systems Became Complex by 1940s–60s

- 1940s: B-29, V-2, Corporal, P-80, P-86
- 1950s: Ballistic Missiles (Atlas, Titan, Polaris, Jupiter, Minuteman), Spacecraft (Explorers and Pioneers), and Aircraft (B-36, B-52, F-104)
- 1960s: Aircraft (F-111), Spacecraft (Mercury, Gemini, Apollo, Ranger, Mariner)

## ◆ Sample Drivers

- Most vehicles don't return from space; those that do must be even more reliable
- Extreme temperatures, pressures, vibration loads
- Heterogeneous technologies (structures, mechanical, electronics, computing, control, rocket propulsion)
- High, concentrated energy (rocket propulsion, reentry)
- Zero gravity (floating particles)
- Autonomy (classical Guidance, Navigation, and Control (GN&C), command sequencing, failure management)



*Complex Defined as “Beyond a Single Person’s Understanding”*

# Systems Integration: Genesis in Failure



- ◆ **Greater than 50% Failure Rates**
- ◆ **Primary Causes**
  - Mismatches between design-to-build and as-flown configuration
  - Vibration damage
  - Debris/contamination
- ◆ **Systems management/integration developed by U.S. Air Force (USAF) and U.S. Army, Jet Propulsion Laboratory (JPL), and Army Ballistic Missile Agency (ABMA) in 1950s-60s for ballistic missiles and early spacecraft**
  - Bernard Schriever (WDD, Ballistic Missile Division)
  - George Mueller (NASA Apollo)
  - Simon Ramo (Ramo-Wooldridge)
  - Samuel Phillips (Minuteman)
  - Jack James (JPL)
  - Wernher von Braun (ABMA, MSFC)



*Samuel Phillips Papers*  
Library of Congress

*By Mid-1960s, Failure Rates Decreased to Less Than 10%*



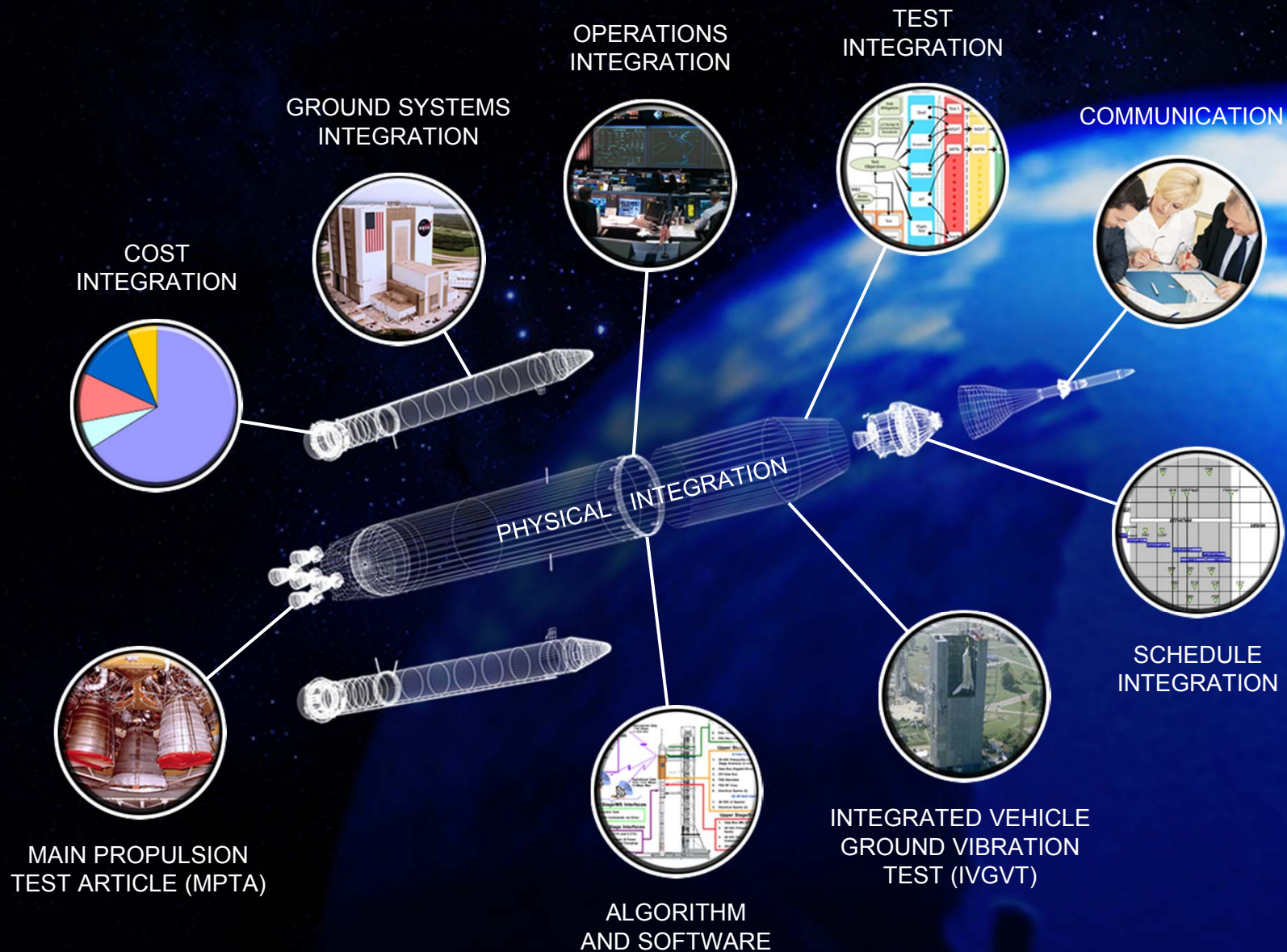
# Systems Management / Integration



- ◆ **Project Management – Manager’s Approach**
  - Heavy pre-planning: System does not return, must build it right the first time, cannot fly and return prototypes (as with aircraft)
  - Organize by product instead of by discipline
- ◆ **Systems Analysis – Scientist’s Approach**
  - Developed from operations research; assess technical feasibility, design, and operational alternatives
- ◆ **Systems Engineering – Engineer’s Approach**
  - Configuration control, coordinate diverse subsystem technologies
- ◆ **Configuration Management**
  - Key link between management hierarchy and engineering working groups
  - Developed at Boeing for Minuteman (Sam Phillips)
  - Ties cost and schedule to engineering changes
  - Design reviews tied to “change freeze and baseline” process
- ◆ **Environmental and System Testing + Reliability and Quality Assurance Methods**
  - Tests: Vibration, Thermal Vacuum, Systems Integration, “Search for Weaknesses”, Failure Injection

*Various Methodologies for the Same End Game: Mission Success*

# Systems Integration: Consistency Theory in Practice

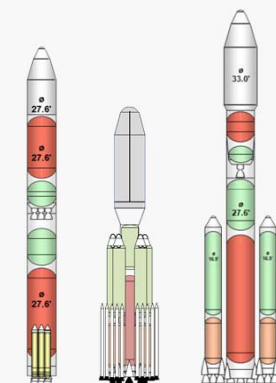


# Systems Integration in Human Space Flight



## ♦ Human Space Flight Values Complexity and Gigantism

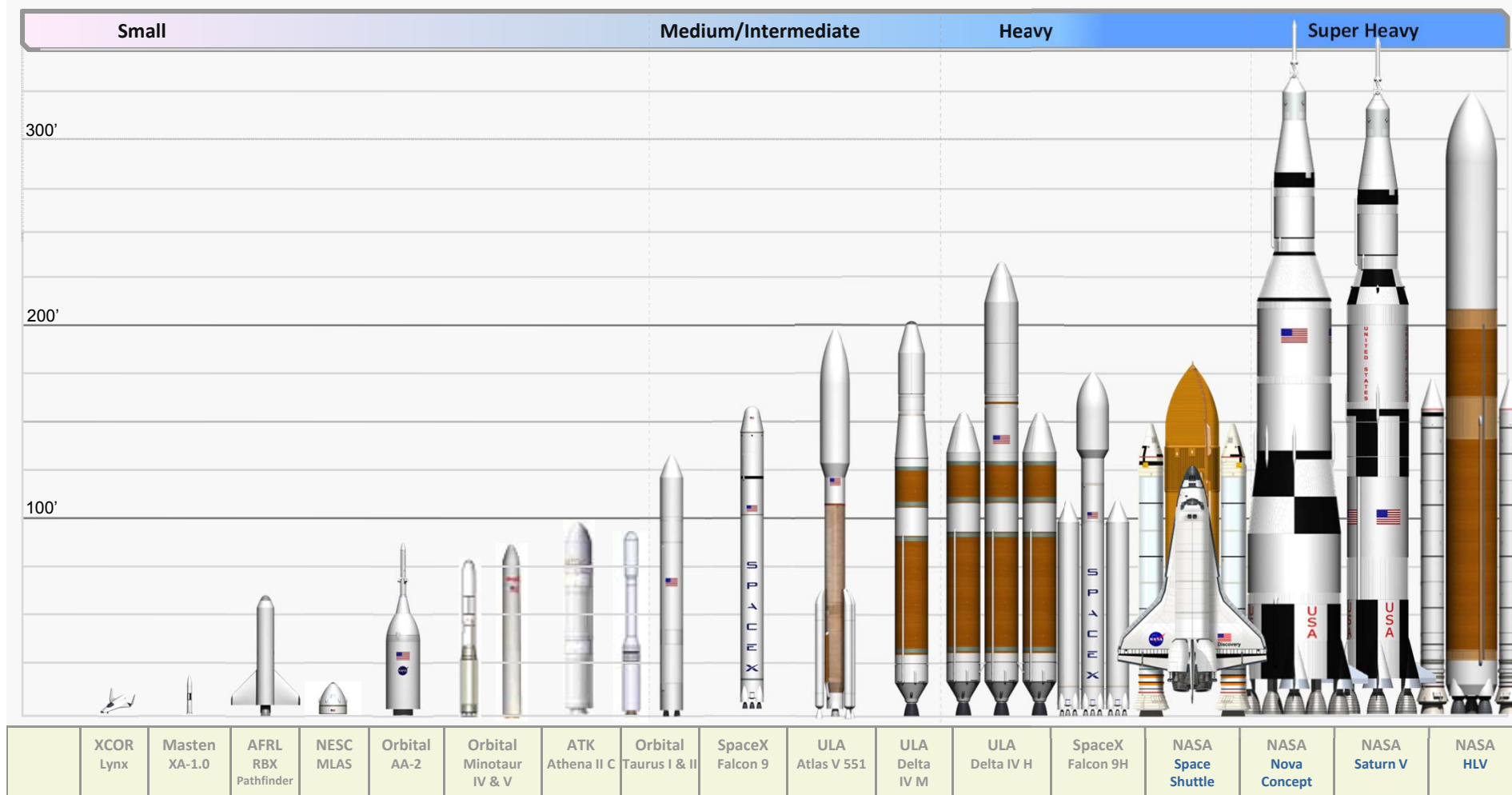
- Crew capsule systems are complex and heavy
- Launch vehicles tend toward gigantic proportions
  - Safely lift the crew capsule in the event of an early engine shut down or other launch emergency
  - Substantial energy necessary to get out of Earth's gravity well and to achieve a sustainable orbit
  - Launch vehicles tend to be more mass efficient when staged



*Mass Drives Cost and Size*



# Spectrum of United States Launch Vehicles



4/8/2011 21st CGSP

*Sample of Proposed and Fielded Systems*

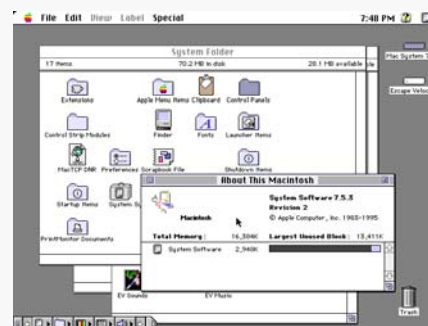
# System Context: The Guiding Principle of Systems Integration



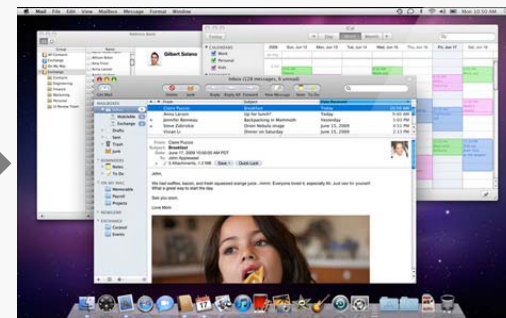
- ◆ Understanding the System Context is essential to a coherent integration approach
- ◆ Maintaining System Integration within the System Context is essential to efficient system life cycle execution
  - Apple's computers have successfully followed a basic paradigm over the past 30 years
  - The original principles of the Macintosh are embodied in today's computers as they were in the original 1984 model
    - Simple user interaction and operation guide the implementation and integration of Apple applications
    - Apple products provide an ever increasing set of complex functions, yet within the same basic System Context and Systems Integration model



1984 Mac Desktop



OS 7.5.6 Mac Desktop



2011 Mac Desktop

Images from <http://en.wikipedia.org/wiki/Macintosh>

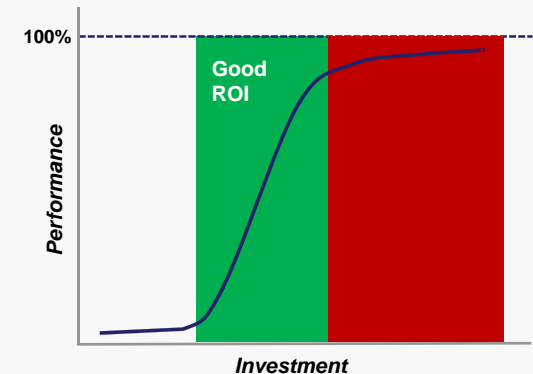
Image from Apple Inc.: <http://www.apple.com/>

*Systems Integration Is Specific to the System Being Integrated*

# Understanding Context: Cost is a Function of Performance



- ♦ Extreme requirements drive the cost 318%
- ♦ Question: Is a 14% increase in maximum speed (performance) worth 318% increase in cost?
- ♦ Question: Is a 34% increase in 0 – 60 mph (performance) worth 318% increase in cost?



*Porsche 911 Carrera*



Cylinders	6
Engine layout	Rear
Performance	180 mph
0-60 mph	4.7 sec
MSRP	\$77,800



Horsepower 345

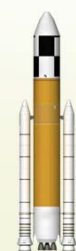
70 MT

OR

*Porsche 911 Turbo*



Cylinders	6
Engine layout	Rear
Performance	195 mph
0-60 mph	3.5 sec
MSRP	\$160,700



Horsepower 530

100 MT

OR

*Porsche 911 GT2 RS*



— Source: Porsche website

Cylinders	6
Engine layout	Rear
Performance	205 mph
0-60 mph	3.4 sec
MSRP	\$245,000



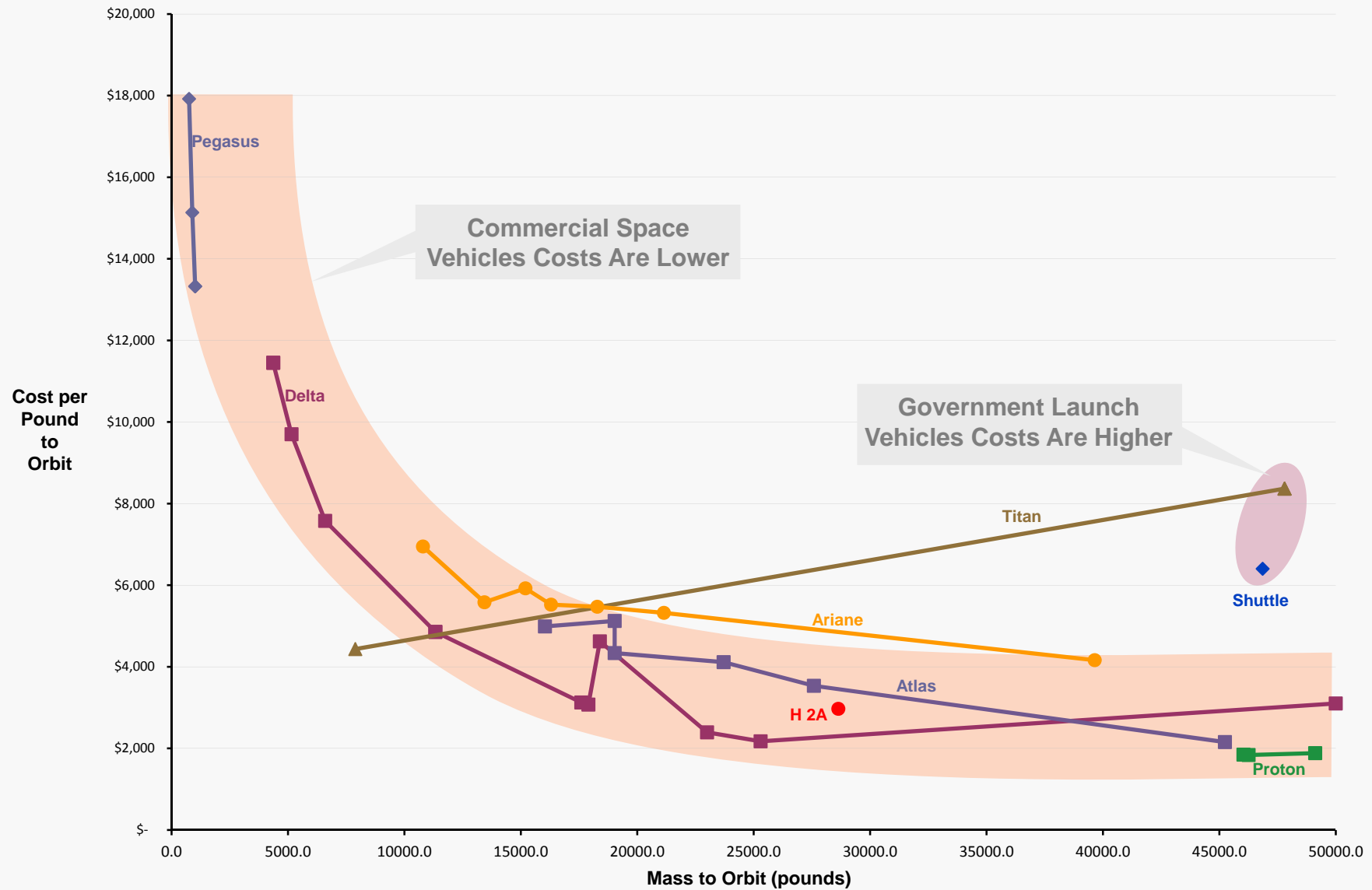
Horsepower 620

130 MT

*System Context Drives Decisions*



# Payload to Orbit Cost Trends



# System Context Provides Direction for Systems Integration



## ◆ Uses

- Human Transport
- Science Missions
- Resupply

## ◆ Operations

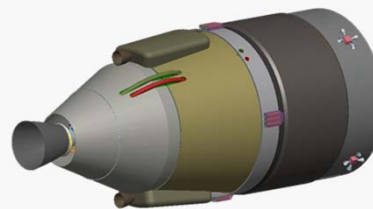
- Autonomous
- Manual
- Human/System Interaction
- On-Board Operations
- Fixed-Base Operations

## ◆ Environments

- Low-Earth Orbit
- Interplanetary Space

## ◆ System Context sets

- Interfaces and Interface Constraints
- Algorithm and Software Integration
- Operations Integration
- Integrated Test and Verification Set



*The System's Use Is Critical to  
Delivering the Right Product for the Purpose*

# Systems Integration and System Architecture



- ◆ **Systems Integration involves defining the System Architecture**
  - Distribution of Functionality
    - System Elements
  - Distribution of Operations
    - Ground Based/On-board
    - Automated/Manual
  - Definition of Assembly Process
- ◆ **System Architecture involves various aspects**
  - Durability – The system should function robustly and remain in good condition
  - Utility – The system should be useful and function well for the users
  - Beauty – The system should delight stakeholders and raise their spirits
  - Based on architectural concepts of Vitruvius  
(<http://penelope.uchicago.edu/Thayer/E/Roman/Texts/Vitruvius/home.html>)
- ◆ **Robustness (durability and utility) of the concept provides ease of handling integration issues**
  - Cost Margins
  - Cost Reserves
  - Performance Margins

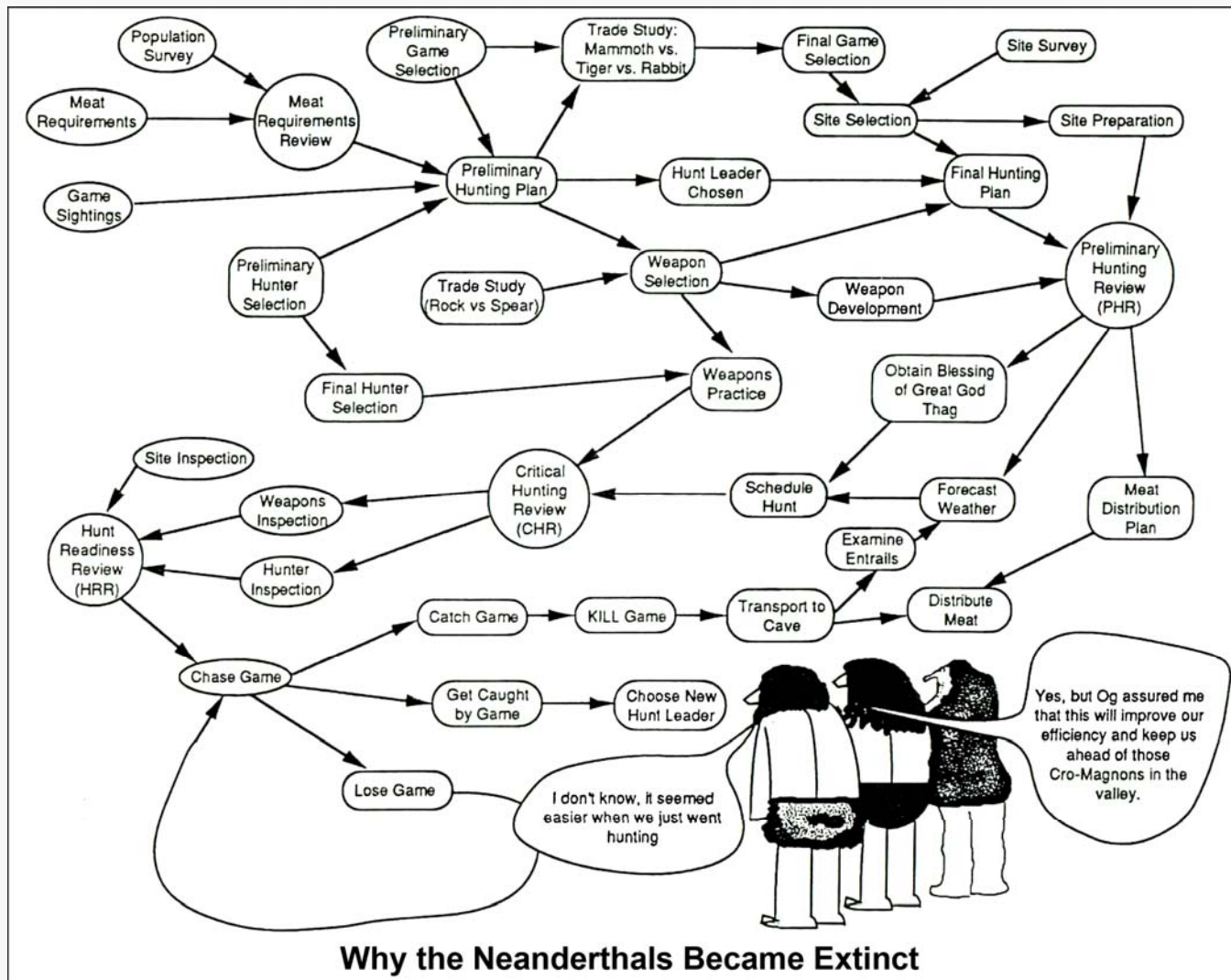


*System Architecture Most Clearly Embodies “Form Follows Function”*

(Louis Sullivan: <http://www.prairiestyles.com/lisullivan.htm>)



# Driving in Complexity: The Path to Extinction



# Beautiful Systems Are Simple



- ◆ **System Integration must stay focused on fostering simplicity in the system interactions, interfaces, and operations**

- System Integration can easily drive complexity into any system interaction, interface, or operation
- Apple provides very robust applications integrated through block upgrades
  - iPod => iTouch => iPhone => iPad



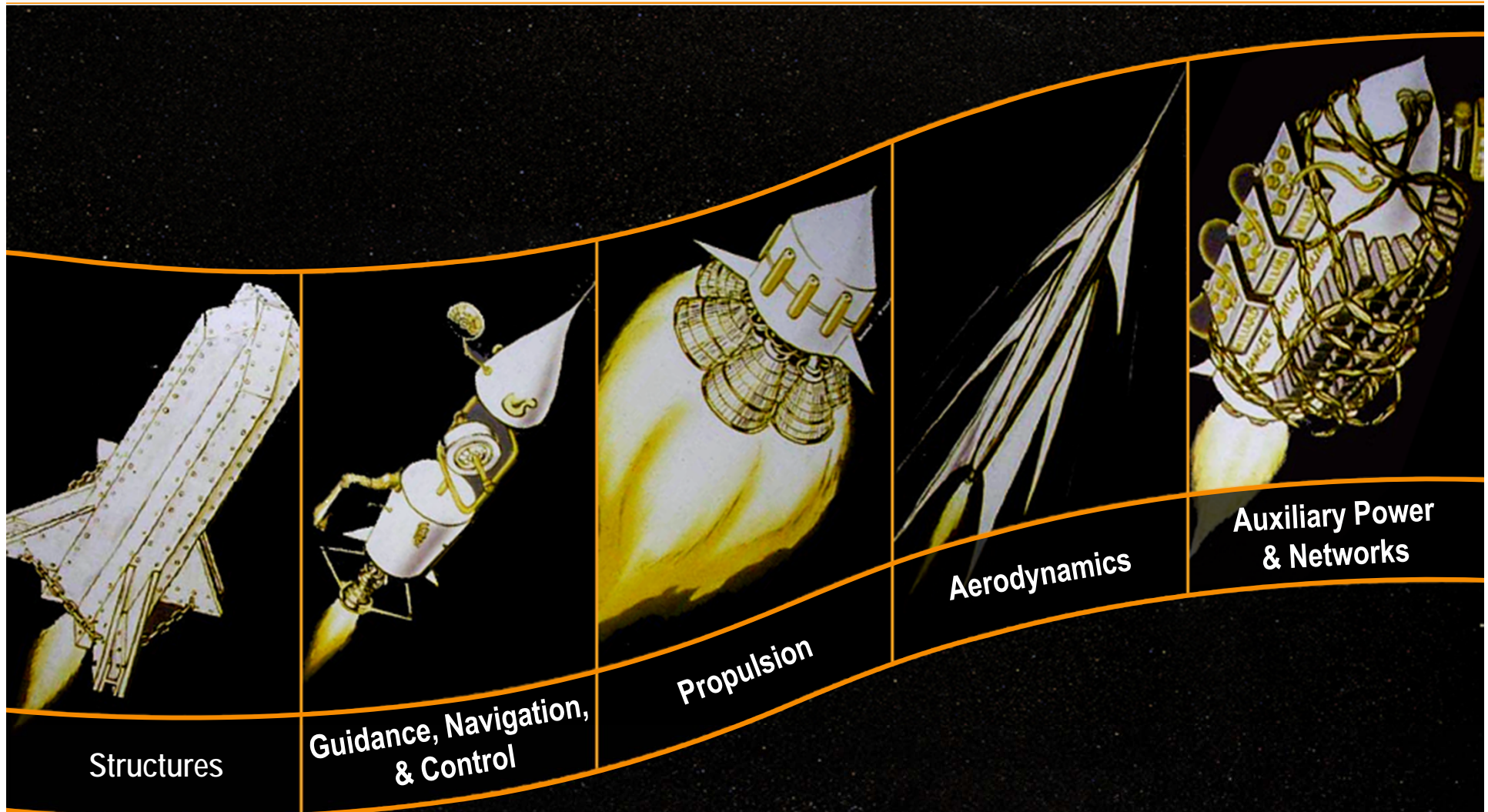
- Each product release incorporates new capabilities that expand and improve user applications and interactions, built on familiar foundations that came before

- ◆ **Block Upgrades provide an approach to manage increasing system functionality and cost, and provides**

- An opportunity for the system to stay focused on a few simple integration activities rather than diverse and complex integration activities
- An opportunity for off-ramps of the system or of system functions as experience is gained with system use and operation
- Manageable steps to increase system functionality over time
  - Timely initial release
  - More robust functionality in subsequent blocks

*Images from Apple Inc.: <http://www.apple.com/>*

# Beautiful Systems Appropriately Integrate Functions



*Every Group Has Their View Of The System. The Integrator Must See the Whole, With Each Facet Appropriately Incorporated.*



# Stability of System Concept



- ◆ **System Concept must be consistent**
  - Technically
  - Financially
  - Schedule
  
- ◆ **System Concept must be persistent**
  - The concept cannot change with time or circumstance
    - Changes involve a redefinition of the system and the system expectations = a New System
  
- ◆ **System Integration is dependent on sustaining the system concept through the development and operations phases**

*Systems That Are Consistent and Persistent Are Stable*

# Communication: Integrating the Integrators



## ♦ Effective communication is essential to integration

- All groups (Design, Test, Operations, Maintenance) must be working to the same objectives in the same context
- Multiple, uncoordinated approaches lead to multiple solutions to the same question or challenge
  - The source of the greatest defeat of system simplicity
  - Greatly increases integration complexity
- Groups working different aspects of a solution must be maintained in sync
  - Communication is the key, *not process*
    - Process facilitates communication, but will not maintain sync by itself
    - Communication pathways should be simple
      - Organizational stove pipes inhibit communication and must be explicitly managed
      - Functional swim lanes help clarify responsibility and must be actively managed to avoid becoming barriers to communication
  - Communication is personal

*Communication Is the Framework in Which System Integration Takes Place*

# Summary



- ◆ **System Integration was refined through the complexity and early failures experienced in rocket flight**
- ◆ **System Integration encompasses many different viewpoints of the system development**
- ◆ **System Integration must ensure consistency in development and operations activities**
- ◆ **Human Space Flight tends toward large, complex systems**
- ◆ **Understanding the system's operational and use context is the guiding principle for System Integration**
  - Sizeable costs can be driven into systems by not fully understanding context
  - Adhering to the system context throughout the system's life cycle is essential to maintaining efficient System Integration
- ◆ **System Integration exists within the System Architecture**
- ◆ **Beautiful systems are simple in use and operation**
  - Block upgrades facilitate manageable steps in functionality evolution
- ◆ **Effective System Integration requires a stable system concept**
- ◆ **Communication is essential to system simplicity**



# Exploring Space for America's Future



National Security



Public Engagement



Scientific Knowledge



**New  
National  
Capability**

Economic Prosperity



Technology Development

Global Partnerships





**For More Information**



**[www.nasa.gov/exploration/systems](http://www.nasa.gov/exploration/systems)**